

# INFO-F-404 : Real-Time Operating Systems

## Project 1 : Audsley

R my Detobel, Stanislas Gueniffey and Denis Hoornaert

December 4, 2017

### Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Implementation choices</b>	<b>1</b>
<b>3</b>	<b>Difficulties encountered</b>	<b>1</b>
<b>4</b>	<b>Project output dependencies</b>	<b>2</b>

### Abstract

## 1 Introduction

## 2 Implementation choices

## 3 Difficulties encountered

One of the difficulties encountered during the realisation of the project is the generation of random systems. More precisely : how to ensure that the utilisation of the generated system is as close as possible to the utilisation specified by the user ?

The issue has been addressed as follow :

By definition, the total utilisation of the system is given by :

$$U(\tau) = \sum_i^n U(\tau_i) \qquad U(\tau_i) = \frac{C_i}{T_i}$$

Where :

- $n$  is the number of tasks specified by the user
- $U(\tau)$  is the total utilisation of the system specified by the user

We generate a vector of size  $n$  for which each cell contains a random integer  $i \in [10, 100]$ . This vector is supposed to be the vector of tasks utilisation. Thus, we compute the factor by which each cell has to be divided so that the sum is equal to the desired utilisation.

$$U(\tau) = \frac{\sum_i^n rand(i)}{factor} \qquad \Leftrightarrow \qquad factor = \frac{\sum_i^n rand(i)}{U(\tau)}$$

For example :

$$rand = [30, 40, 50, 60], \quad |rand| = 4, \quad U(\tau) = 70$$

$$factor = \frac{30 + 40 + 50 + 60}{70} \approx 2.57$$

$$utilisations = [11.67, 15.56, 19.46, 23.35]$$

Once done, we can create a task based on the given utilisation (one of the vector) as follow :

1. we generate a random offset
2. we generate a random wcet with the condition that it must be greater than 0 because otherwise the task is useless and because a negative wcet makes no sense
3. for the period, we cannot generate a fully random value as the utilisation must be respected. Hence, we determine its value using the following formula :

$$U(\tau_i) = \frac{C_i}{T_i} \quad \Leftrightarrow \quad T_i = \frac{C_i}{U(\tau_i)}$$

4. we generate a random deadline with the condition that it must at least be equal to the wcet and at most equal to the period

Applying this process, implies the use of a lot of float numbers and the approximations that comes with. However, we still manage to have a nice total utilisation approximation.

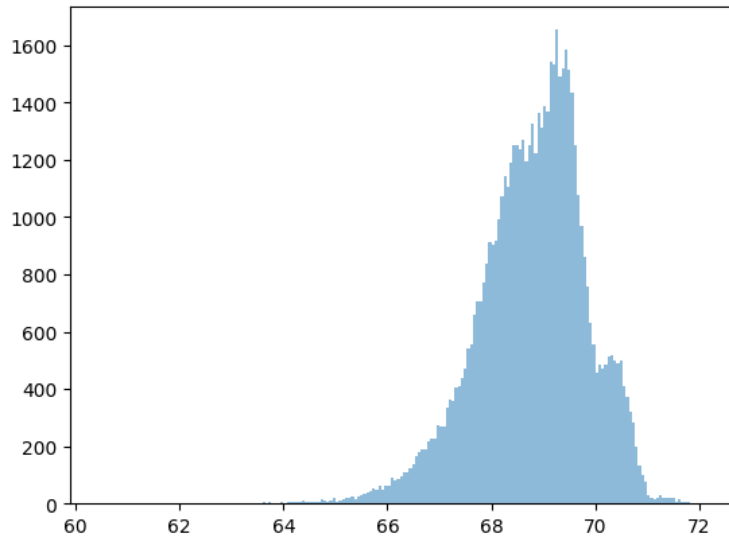


Figure 1: Distribution of utilisation of generated systems (6 tasks with a targeted 70% of utilisation)

## 4 Project output dependencies

The python's modules used to generate the image of the scheduling are called `numpy` and `matplotlib`. Even though they are quite famous amongst python programmers, they are not part of the standard library. Hence, they need to be manually installed sometimes. This can be easily be achieved by using `pip3` and by typing the following commands :

```
sudo pip3 install numpy
sudo pip3 install matplotlib
```

The commands above have been tested on Ubuntu 16.04 LTS but might differ depending on the distribution and the version.